

Lecture Notes for CS 509  
Foundations of Computer Science  
Lecture 11

Lecturer: Joe Kilian  
Scribe: Begumhan Turgut  
Department of Computer Science  
Rutgers, The State University of New Jersey

## 1 Direct Proof Systems:

### Model 1

As shown in Figure 1. Polynomially. The answer is also polynomial in size, this assumption will change in a little bit, they go back and forth.

$$\begin{aligned} Q_i &= \text{Function of } x, R, Q, A_1, Q_1^1, A_1^1, \dots, Q_{i-1}^1, A_{i-1}^1 \\ \text{if } x \in L(P_1, P_2, V) &\text{ accepts w.p. } 1 \\ \text{if } x \ni L(\hat{P}_1, \hat{P}_2, V) &\text{ accepts w.p. } = < 1/2 \\ \forall \hat{P}_1, \hat{P}_2 & \\ \text{accept } P(x, R, Q_1, A_1, \dots, Q_k^1, A_k^1) & \end{aligned}$$

Another option is the probability of catching a bad prover; which than would have the following :

$$\begin{aligned} \text{if } x \in L(P_1, P_2, V) &\text{ accepts w.p. } \geq 3/4 \\ \text{if } x \ni L(\hat{P}_1, \hat{P}_2, V) &\text{ accepts w.p. } (1 - 1/|x|O(1)) \end{aligned}$$

### Model 2

Only allow 1 question to each prover. Just have a couple of questions each, all of the researchers' experiences should tell us that the system in Model 1 is much more powerful

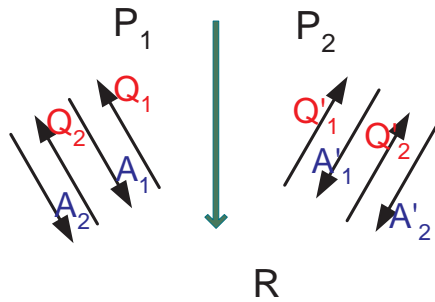


Figure 1:

than the system in Model 2.

Accept  $P(x, R, Q_1, A_1, Q_1^1, A_1^1)$

If I catch you, than I stop and say I dont believe you anymore.

There is a reason to believe that I can implify this without asking bunch of questions, and getting answers.

### Model 3

Now, let us look at a completely different model next, which has nothing to do with provers, as shown in Figure 2 (a). In this model, called The Oracle Proof Model, we write down the proof, back to NP, except the proof might be exponential size, gigantic. So, how powerful this proof will be? If we have an exponential time verifier than it is powerful, but if we have a polynomial time verifier, than it will get tired, so lets give this verifier a fighting chance.

V has random access memory | it is called an oracle proof.

This would be thought of as very powerful but no, it is still NP, deterministic, we still have polynomial size proof.

The two variations of this, as shown in Figure 2 (b)

V can check  $poly(|x|)$  locations

V can check 2-blocks of locations

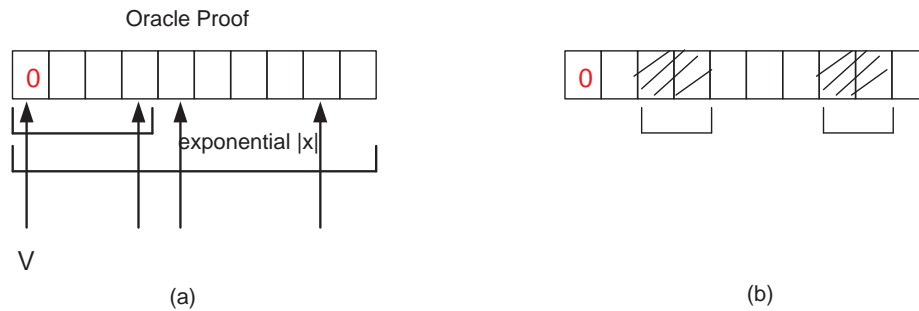


Figure 2:

## 2 MIP $\longrightarrow$ Oracle Proof (can be converted as follows)

**Lemma:** WLOG  $\hat{P}_1, \hat{P}_2, \dots, \hat{P}_k$  can behave deterministically. This is true for both good and bad provers. If all your goal is to have verifier accept, than you might as well be deterministic.

Migh as well use common pool of coins,  $R_p$  (choice of random coins). Let us also point out that coins are random variables, and provers are not flipping coins.

$$\begin{aligned} Pr[(P_1, V) \text{ accepts}] &= E_{R_p}(P_1, \dots, P_k, V) \text{ accept using } R_p \\ &\longrightarrow \exists R_p Pr[(P_1, V) \text{ accepts using } R_p] \geq E() \end{aligned}$$

Lets make a note that it is important to remember that you cant allow  $P_1$  s strategy to chance once it is written down, based on something  $P_2$  sees.

Deterministic strategy means function F  $P_i$  s view goes to  $P_j$ s Answer.

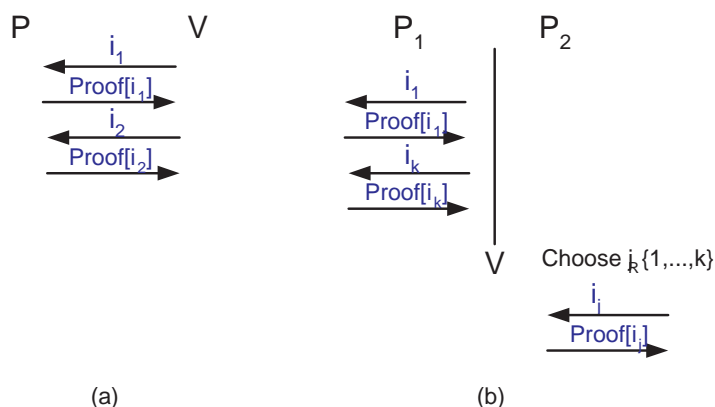


Figure 3:

Now, given a strategy, how do we convert that to an oracle proof?

Write down all strategies of all  $P_i$ s (provers strategy book).

This is merely exponential size oracle proof.

$V_{op}$  simulates V interacting with  $P_1 \dots P_k$ .

Probability of accepting should be exactly the same, if they are good provers, or if they are bad.

### 3 Oracle Proof $\longrightarrow$ P

L, whenever V wants to check Proof [i] (i can be polynomially many digits number).

Let us look at Figure 3 (a). According to this system, we could ask; are we done?

No, what if I am dishonest!

There is no commitment, I have not committed ahead of time how this proof should look like, I can change it on the fly.

How will you make me commit?

So, we throw in another prover, as shown in Figure 3 (b)(they, the Ps, still cant talk to each other), the verifier keeps talking to  $P_1$ .

$P_1$  is seeing all these questions, and getting an idea of what the verifier wants to see, so  $P_1$  shapes its answers accordingly.  $P_2$  cant do the same based on seeing one-question, so it

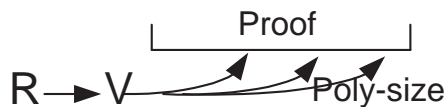


Figure 4:

doesn't have any context for the questions.

$V$  accepts iff

would have accepted Proof as told to by  $P_1$

AND

$P_1, P_2$  agree on value of  $\text{Proof}[i_j]$

What if  $x \in L$ ,  $\hat{P}_1, \hat{P}_2$  (They might have randomized strategy but equally good!)

WLOG:  $\hat{P}_1, \hat{P}_2$ , deterministic.

For any  $i$ ,  $\hat{P}_2$  gives fixed value for  $\text{Proof}[i]$ ; this is an oracle proof. Even though, it has never been written down, the behavior of  $P_2$  shows an oracle proof.

Now, there are 3-cases:

1)  $V$  would have accepted Proof (this is  $\hat{P}_2$ ).

2) with Prob  $\varepsilon$ ,  $V$  rejects Proof.

2a)  $P_1$  behaves consistently.

$V$  rejects.

2b)  $P_1$  inconsistent on  $\geq 1$  out of  $k$

Gets caught with probability  $1/k$

## 4 How powerful is MIP:

$\text{PSPACE} \subseteq \text{MIP} = \text{NEXPTIME}$

Suppose we want to verify polysized proof with small number of questions as shown in Figure 4. This is same frame work as the oracle.

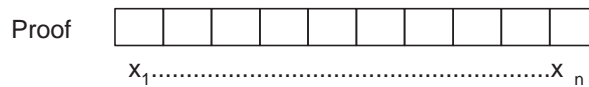


Figure 5:

**Probabilistically Checkable Proofs;** as shown below.

PCP (num of random bits, num of queries, size, error) ← There are lots of number of parameters!

NP  $\subseteq$  PCP (0, poly, poly, 1/poly)  
 ( $O(\log n)$ ,  $O(1)$ , poly,  $(1 - 1/poly)$ ) ← Is this doable? This, we claim is also easy!

$$F = (x_1 \vee \bar{x}_2 \vee x_7) \wedge \dots (x_3 \vee x_5) \text{ As shown in Figure 5.}$$

**PCP Theorem says:**

NP  $\subseteq$  PCP ( $O(\log n)$ ,  $O(1)$ , poly, 1-Omega(1)).

Last parameter refers to a constant chance of catching a "bad proof".

As shown in Figure 6 (a), we have again, 2-questions, 2-answers, have a constant chance of catching a false statement.

Figure 6 (b) shows what is called the Sequential Repetition. Doing sequence of games, one after the other with single prover, and single verifier.

In Figure 6 (c), we give the prover a little lay away. It gets to choose  $Q_2$ , knowing  $A_1$  but it doesnt help. This is called Parallel Repetition with single prover, and single verifier.

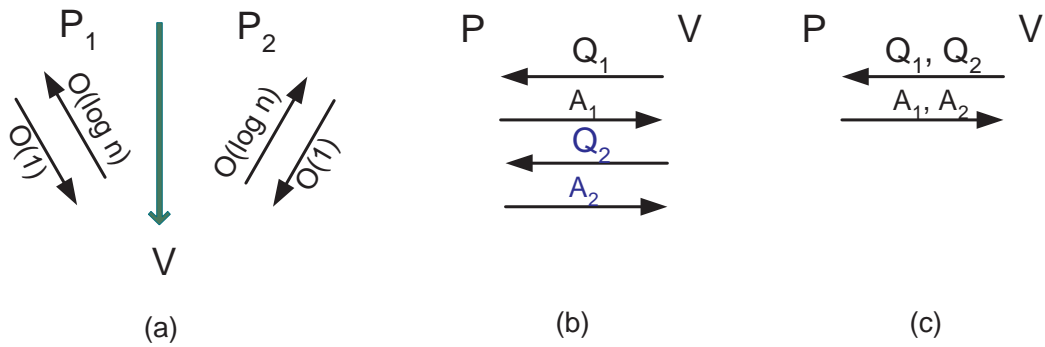


Figure 6:

What about 2-prover proof systems?

We play one game, if I accept then we play another game, I superscript by the game, this will get my error down exponentially by the number of time you do the sequence, as shown in Figure 7 (a).

What if we run this in parallel, as shown in Figure 7 (b), in which case, we hope it works as well as the sequential system! But it does not!!!

As shown in Figure 8 (a), What is the probability of both of them winning this game? About at most  $2/n$ .

V accepts iff  $A_1 = y$  or  $A_2 = x$ .

Suppose we play the game twice in parallel as in Figure 8 (b).

V accepts iff  $A_1 = y_1$  or  $B_1 = x_1$

AND  $A_2 = y_2$  or  $B_2 = x_2$

Now, what is the probability? You still dont know the other persons answer/question.

$A_1 = A_2 = x_1$

$B_1 = B_2 = y_2$

Suppose  $x_1 = y_2 = x_1$  Than we would have the following;

$B_1 = y_1$

$A_2 = y_2$  First, and the second player guesses.

So the probability is not  $(2/n)^2$ , it is more like  $1/n$ .

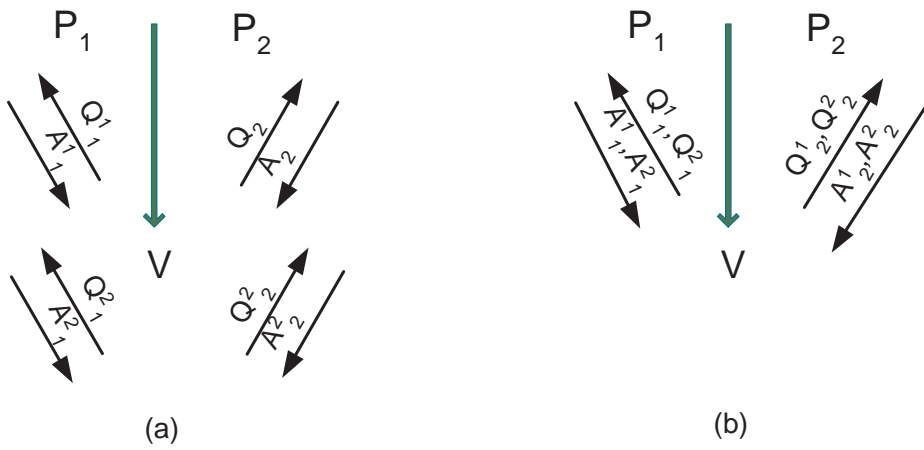


Figure 7:

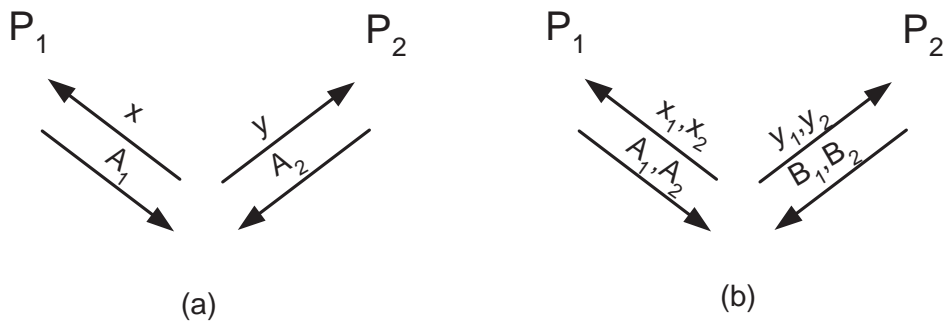


Figure 8:

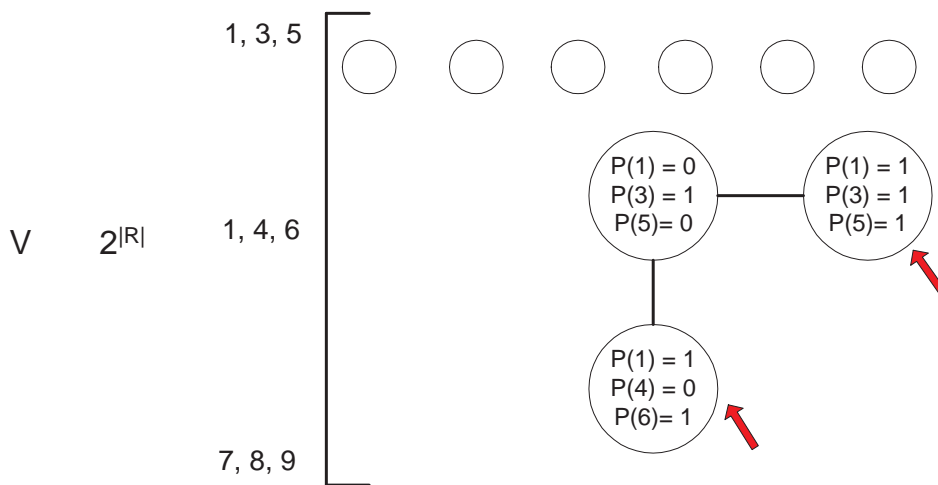


Figure 9:

## 5 Independent set is NP-Hard

Let us take a look at Figure 9.

It is important to point out that the vertices (Partial views) connected by edge iff views are inconsistent.

Each row of the graph corresponds to set of coin tosses.

Each random coin toss specifies few bits of the proof to verify.

For each row, we say what are the views of that proof. In other words, what will make me happy to see.

At this point, we had already gone from a verifier to a graph. And now, I claim;

$V \longrightarrow G$

Proof  $\longrightarrow$  independent set

All vertices (views) consistent with Proof

Independent Set  $\longrightarrow$  Proof

Looking at Figure 10, we can see that this gives me an independent set of size 2. So this shows that the sets pointed with an arrow in the figure are inconsistent.

Now the interesting thing is as follows;

$\Pr[V \text{ accepts Proof}] = P$

$\iff |\text{independent set}| = P * 2^{|R|}$  (And this goes back and forth)

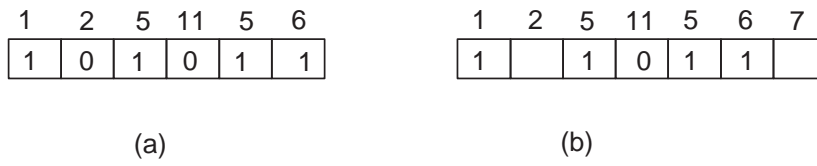


Figure 10:

PCP-Theorem says;

Take  $L \in NP$ ,  $X$  can construct proof s.t.

$N = \text{Poly}(n)$  coin sequences

If  $x \in L$ ,

$\Pr(V \text{ accepts}) = 1, \alpha(G) = N$  (size of the largest independent set)

If  $x \notin L$ ,

$\Pr(V \text{ accepts}) \leq 1/2, \alpha(G) = N/2$